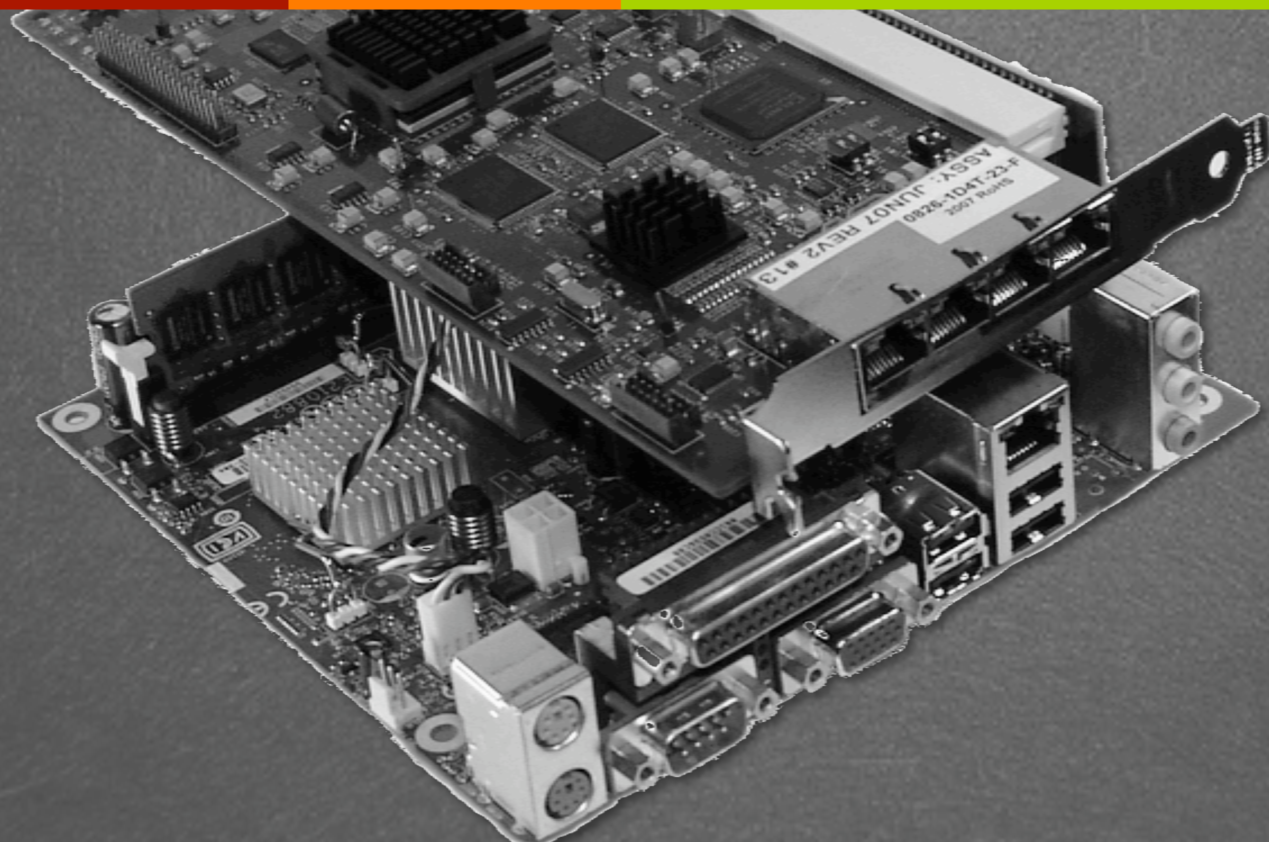
The Axon Network Device: Prototyping with NetFPGA



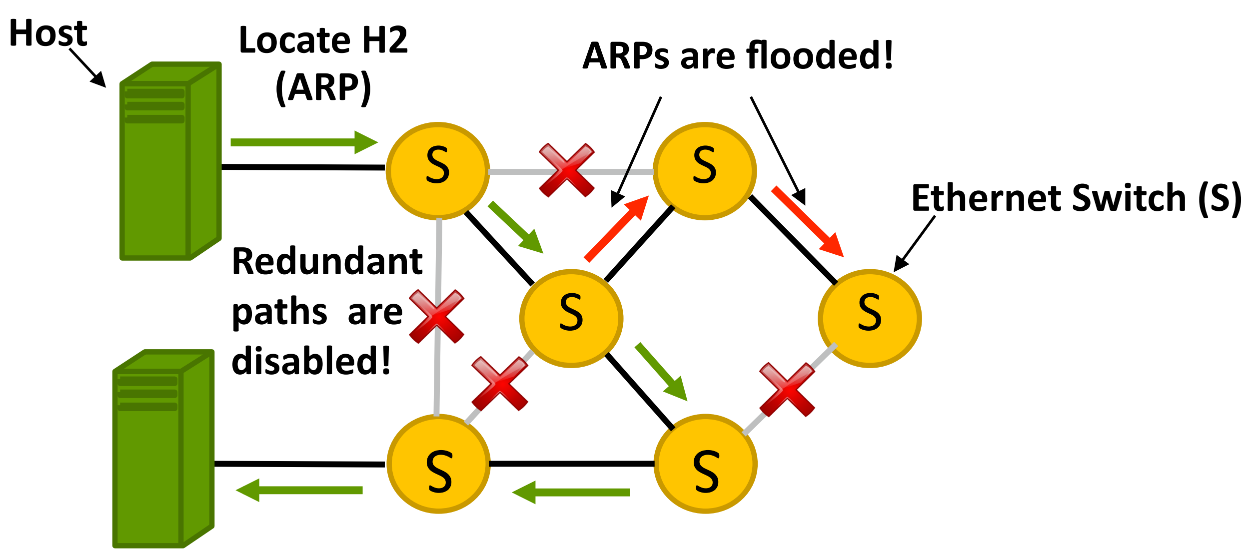
Jeffrey Shafer, Mike Foss, Scott Rixner, Alan L. Cox

Rice University

June 2009

Networks

- Ethernet is a popular "plug and play" network
  - Big challenge – scalability (i.e. broadcast)
  - Big challenge – performance (i.e. spanning tree)



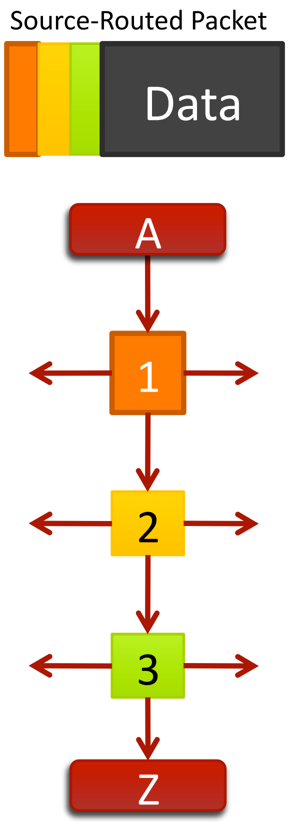
Raising the Bar

- Many researchers have proposed designs to solve Ethernet problems
  - But, they evaluate their solutions in *simulation*
- We have designed a new network device (the **Axon**) that improves Ethernet and have built a **prototype** for evaluation
  - The prototype proves that our design is **practical** and **implementable**

Axon Networks

- Vision for a new network infrastructure
  - Replace Ethernet switches with Axons
  - Remove IP routers from network core
    - Only used to communicate with other networks (their traditional role)
- Hosts communicate with Axons via traditional Ethernet
  - Standard NICs, PHYs, protocols for full compatibility
- Axons communicate with other Axons via a new protocol
  - **Source-routed Ethernet**
  - Uses standard Ethernet physical layer (PHYs, cables)

Source-Routed Ethernet



- Backwards compatibility with traditional Ethernet
  - Network appears to hosts as a giant switched Ethernet segment
- Transparent packet rewriting
  - Imagine sending packet from Host A to Host Z…
  - First Axon on path strips standard header and replaces it with hop-by-hop route list ("source route") used for transport
  - Final Axon restores the original header
- Axon design pushes complexity to the network edge
  - State is stored by edge Axons
    - Axons in network core only process the source routes
  - Routes are determined by edge Axons (connected to host)
    - Enables opportunities for the edge Axons to provide security or virtual networks by carefully choosing routes

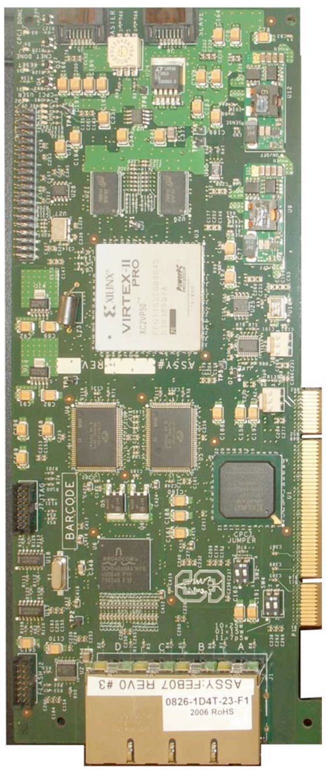
Overview

- Designed the **Axon**, a **new network device**
  - Axons replace Ethernet switches, but are better (performance, scalability, …)
- Built **Axon prototypes** on existing **NetFPGA** platform
  - Easy to use + inexpensive
- Current work – Performance & Backwards Compatibility
  - Demonstrated using Axon prototype
- Future work – Network Scalability (up to 1 million hosts)
  - Intend to demonstrate using a hybrid prototype/simulator infrastructure

Why Prototype?

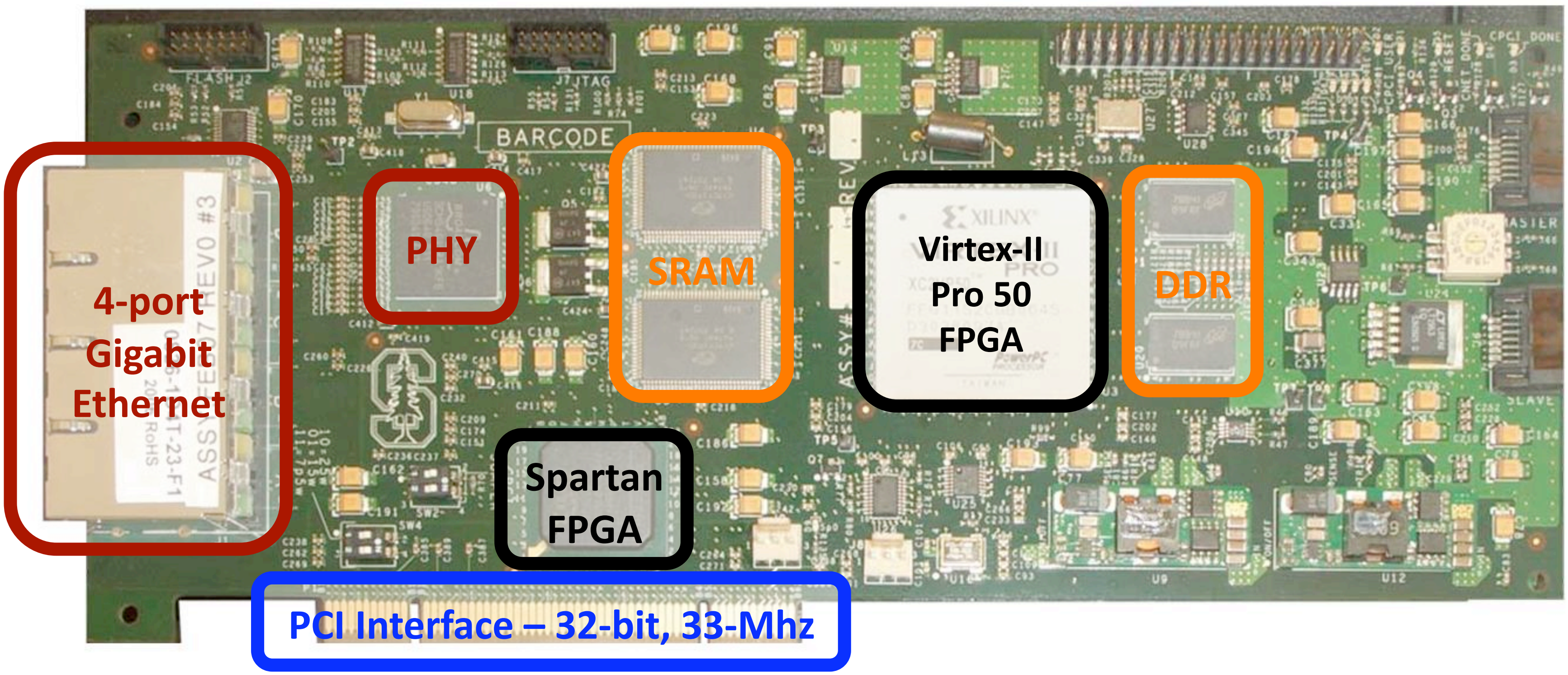
- Compelling demonstrations
  - Does the Axon architecture provides full compatibility with unmodified hosts?
  - Can we translate between traditional Ethernet and source-routed Ethernet at full network speed? *(while using inexpensive / practical hardware?)*
- Compelling experiments
  - How does the performance of Axons compare to Ethernet switches and IP routers when using real network hosts?

Prototype Platform - NetFPGA



- Integrated development environment for network systems architecture research and education
- Developed by team at Stanford
- Inexpensive ($499 academic price)
- Includes PCI board and FPGA reference designs
  - 4-port NIC, Ethernet Switch, and IP router
  - Designs provide all software and hardware components, including Verilog and ModelSim libraries

NetFPGA PCI Board

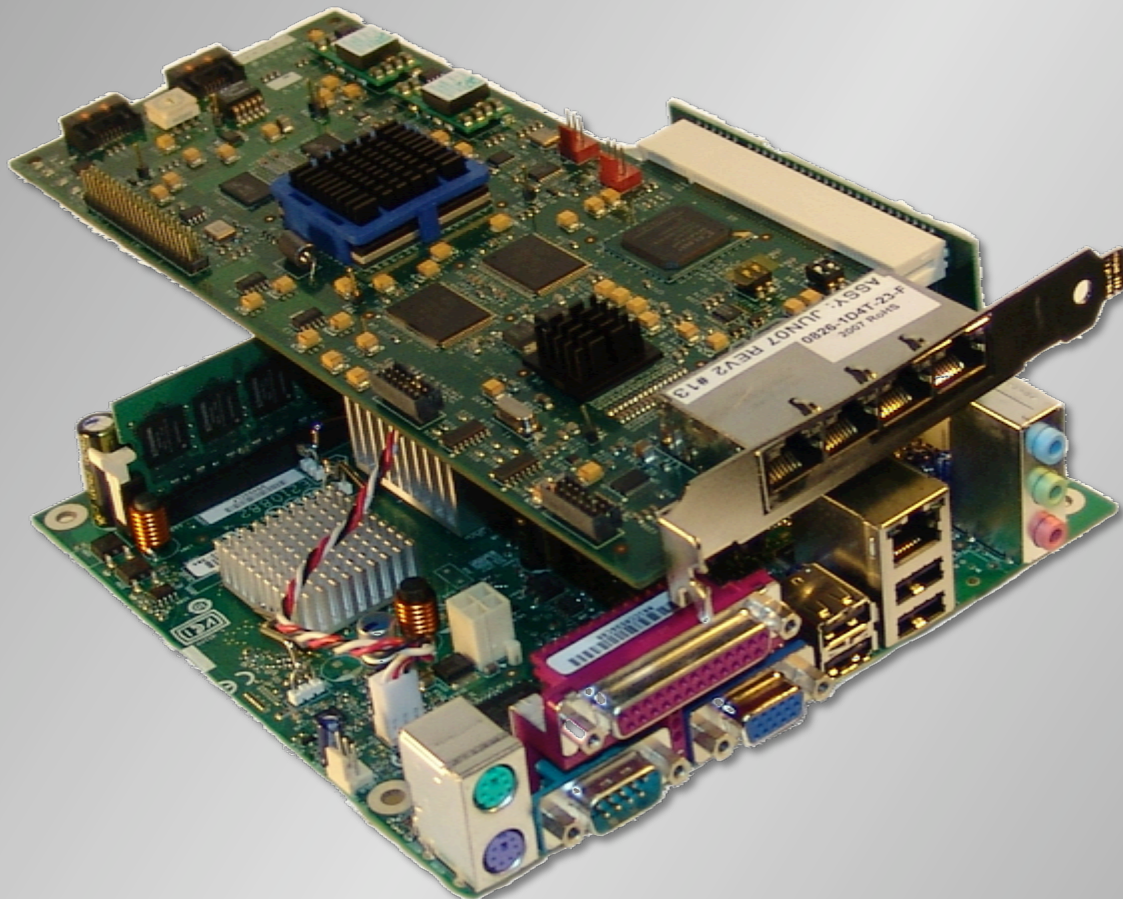


Prototype Architecture - Data Plane

- Data Plane - High-speed packet forwarding on FPGA
- Reused existing NetFPGA components
  - PCI card
  - Verilog design library (e.g. DMA transfer engine)
  - ModelSim test suite
  - FPGA programming utilities
- Added new modules to FPGA (14,000 lines of Verilog)
  - Cut-through packet switching engine
  - Input/output port units to transparently convert between traditional and source-routed Ethernet at line rate

Prototype Architecture - Control Plane

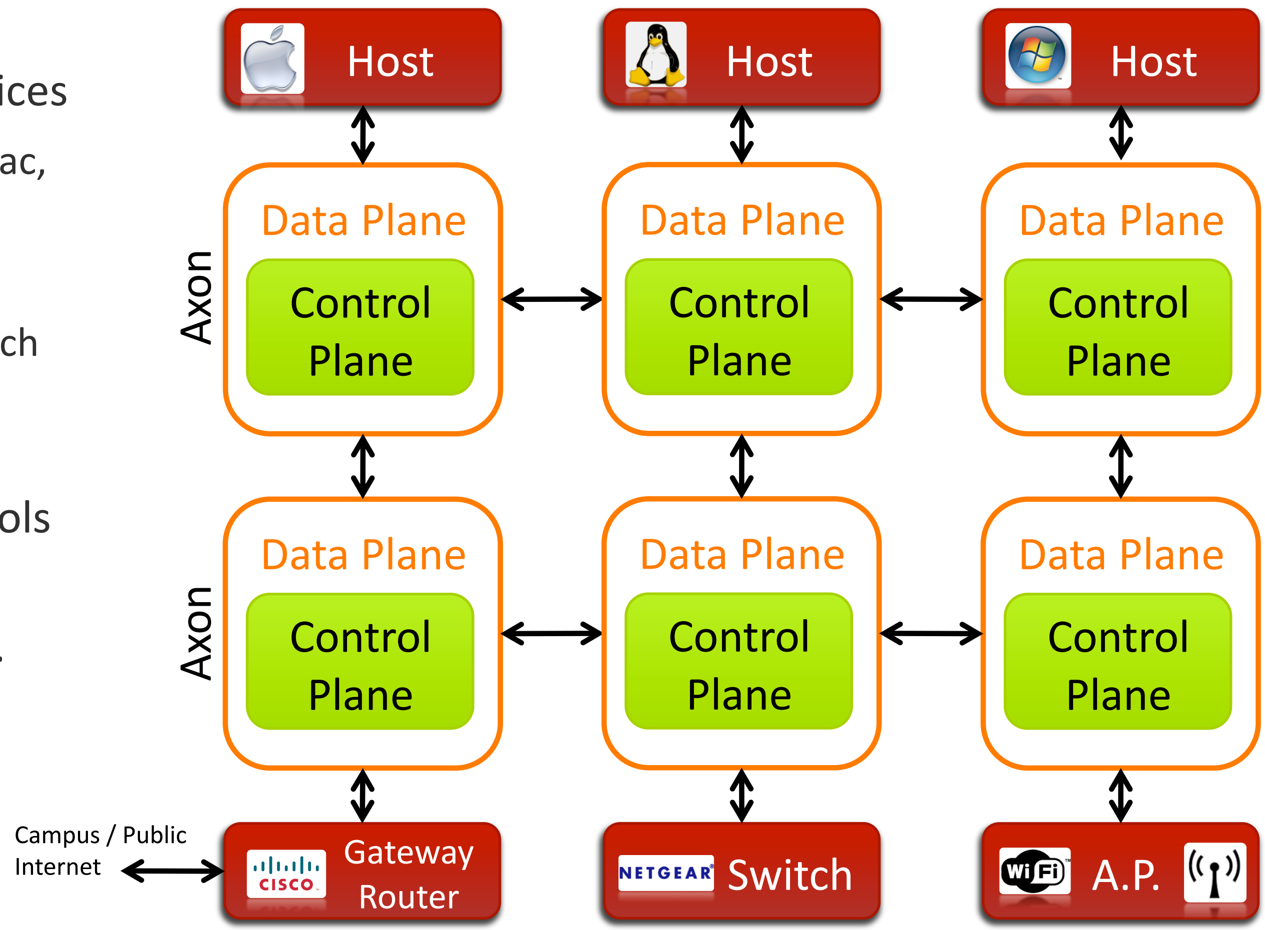
- Control Plane - Management tasks on processor
- Intel Atom motherboard
  - Low-power + low-cost + x86 compatibility
  - Don't need a fast processor for control plane
- Reused existing NetFPGA components
  - Linux (x86) device driver
  - Programming and management tools
- Wrote new 6,000 line control program in C
  - Determine network topology
  - Calculate and establish source-routes
  - Manage Axon hardware



Axon Prototype

Prototype Network

- Unmodified devices
  - Windows, Mac, Linux hosts
  - Wireless A.P.
  - Netgear switch
  - Cisco router
- Standard protocols
  - ARP, DHCP, Ethernet, …
- **Transparent compatibility!**

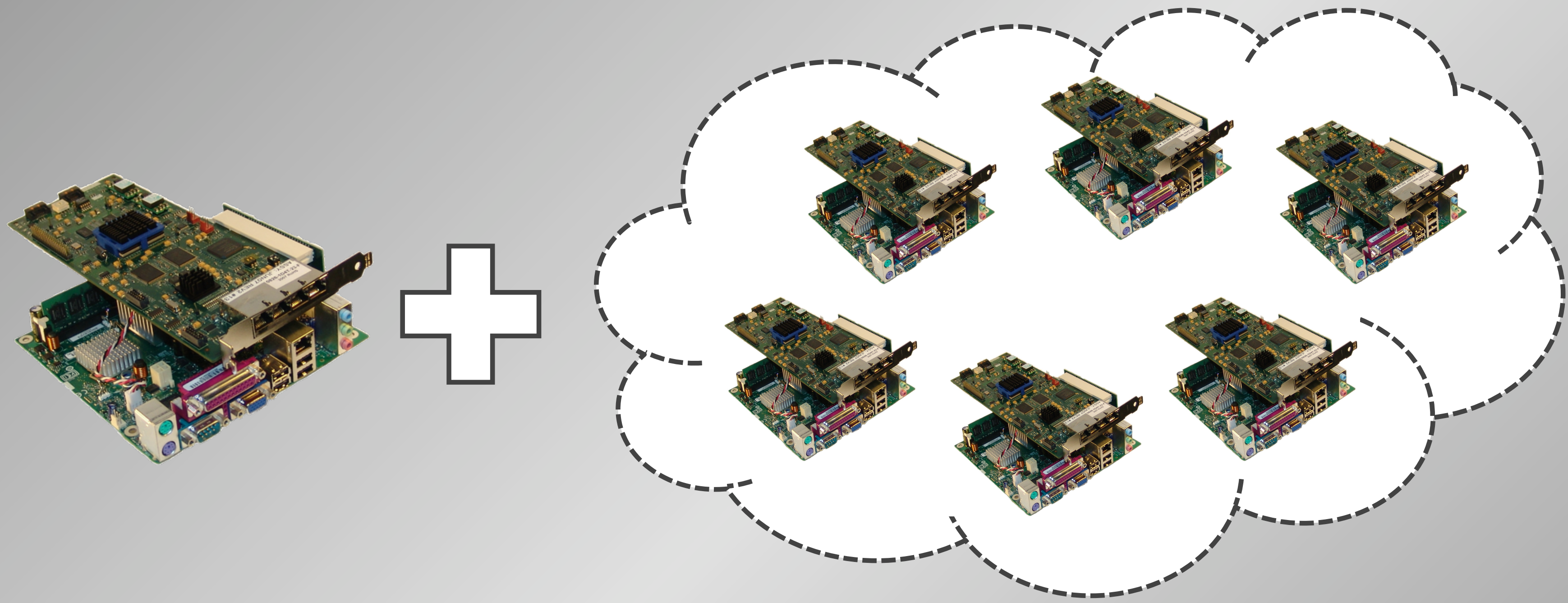


NetFPGA Advantages

- Rapid development
  - Leverage the Verilog library, ModelSim environment, and proven hardware
  - The Axon prototype was a summer project
- Portability (x86-based control software)
  - Leverage this advantage for future simulator work

NetFPGA Drawbacks

- Only 4 Ethernet ports – limits topologies
- FPGA is 76% full
  - Prototype limited to 16-entry CAM
    (CAM is used to translate between destination MAC address and source route at edge Axon)
  - To show scalability on large networks, a much larger CAM would be needed (too big for FPGA)
    - Have alternate design for prototype that uses a direct lookup method based on MAC address

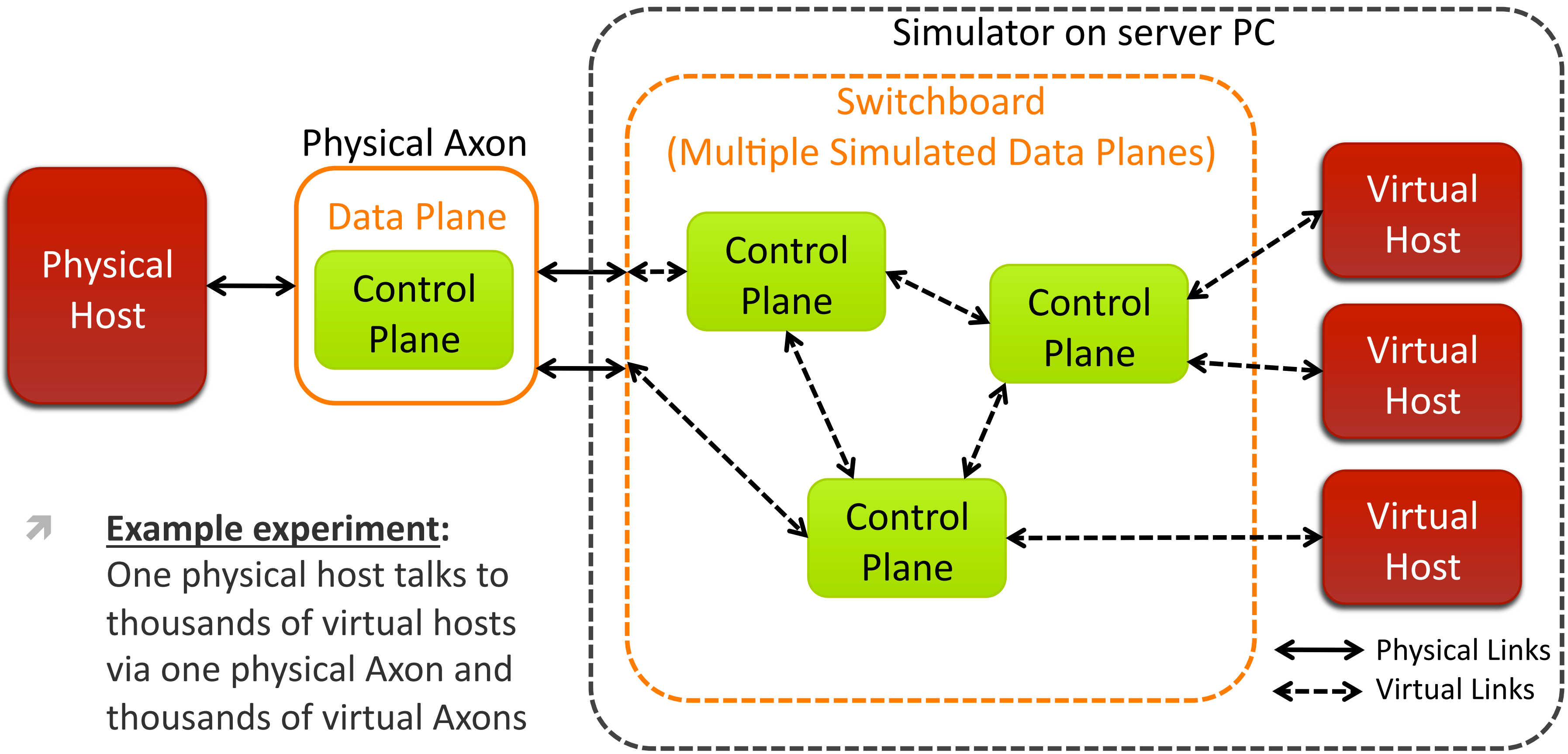


Axon Hybrid Simulator

Network Scalability

- How can we show that the Axon design scales to networks with 1 million+ hosts?
  - Build / deploy vast numbers of prototypes?
- Propose a hybrid prototype / simulator infrastructure instead
  - Use a few real hosts and real Axons
  - Connect them to thousands of simulated Axons (which run the same x86 control software)

Hybrid Infrastructure



- **Example experiment:** One physical host talks to thousands of virtual hosts via one physical Axon and thousands of virtual Axons

How Does This Work?

- Axons store state at edge of network
  - Thus, many interesting metrics are at the edge of the network and can be measured on the prototypes
- What do I use physical Axons for?
  - Initial verification of simulator (inter-operability and correctness)
  - Measure timing overhead to calibrate simulator
    - How long does a route lookup take?
  - Measure scalability of control software under real load
    - Is an Atom processor sufficient to manage thousands of routes?

How Does This Work?

- What do I use virtual Axons for?
  - Measure how many bytes are sent/received to converge to a routing decision
  - Measure how the control overhead increases compared to the network size
  - Determine if the chosen routes efficiently traverse the network
  - More interesting network topologies (physical Axons only have 4 ports)
- Simulator machines will be memory-limited, not processor-limited
  - Envision a dozen+ servers with 24GB RAM each

Axon Prototype Conclusions

- Fully Compatible
  - … with existing hosts, switches, and routers
- Practical Implementation
  - … demonstrated network performance at line rate
  - Competing network designs only use simulation
- Demonstrable Scalability
  - Hybrid prototype / simulation infrastructure
- Thanks to Stanford NetFPGA development team – **http://netfpga.org**

Questions?